I am writing this tutorial to show you how to create a **RESTful** web-service with **PHP 7 & MySQL 8** in a very short time. You can think of this post as a primer. I’ll try to explain the core topics as much as possible while avoiding generic things.

Please note that mysql extension for PHP has been completely discontinued in PHP 7. That is why you can only use mysqli extension. My entire API is based on mysqli extension only.

This tutorial assumes that you know: -

* What an API & REST means.
* What is a REST client.
* What is a RESTful service.
* How to set up a web-server. I am using Apache/2.4.34.
* How to install the necessary Apache modules.

**Application description**

It is a very simple **To-Do** app that is completely based on REST architecture & doesn’t have any GUI. You create or retrieve your To-Do’s using the REST API only.

Newly created To-Do’s are stored in a MySQL database.

I have kept this API extremely simple on purpose. My sole aim here is to get you (& me) started with writing an API & interacting with it. I have skipped database security intentionally & will modify the code later. This tutorial is all about knowing how APIs are created & how you can see REST in action.

**Requirements**

* PHP 7
* MySQL 8
* REST Client such as [Postman](https://www.getpostman.com/). cURL or your favorite browser can also be used.

I am using PHP 7.1.23, MySQL 8.0.16 & Postman for this tutorial. You can use any REST client of your choice. I like Postman better as it makes it very easy & convenient to work with REST.

**Steps to perform**

**1. Create a Database & table**

Create a database & table in MySQL to store the data.

**CREATE** **DATABASE** **IF** **NOT** **EXISTS** **`**my\_to\_do\_db**`**;  
**USE** my\_to\_do\_db  
*--  
-- Table structure for table `my\_to\_do\_tb`  
--***CREATE** **TABLE** **IF** **NOT** **EXISTS** **`**my\_to\_do\_tb**`** (  
 **`**task**`** **text** **NOT** **NULL**,  
 **`**date**`** **text** **NOT** **NULL**,  
 **`**priority**`** **text** **NOT** **NULL**  
) **ENGINE=InnoDB** **DEFAULT** **CHARSET=**latin1 ;

**2. Create**config.php**script**

This PHP script will store database connection-related information. I have kept database credentials in a separate file to-do.ini to avoid hard-coding the password. This file resides in a directory one level above the DocumentRoot. You can put this file anywhere. Just make sure that you refer to the correct location in config.php script.

*config.php script references to-do.ini file.*

**<?php**  
$config **=** parse\_ini\_file('/Users/admin/Sites/to-do.ini');  
$conn **=** mysqli\_connect($config['dbhost'], $config['username'], $config['password']);  
mysqli\_select\_db($conn, $config['db']);

to-do.ini

*username=*

*password=*

*dbhost=*

*db=my\_to\_do\_db*

*Please check my GitHub repository (linked further below) for more information about this file & its location.*

**3. Create a PHP script**add-to-do.php**to add To-Do's**

Now write a PHP script that will put the data in the MySQL table. The idea here is to take three values viz task, date & priority as payload & POST it to the database.

**<?php**  
**include\_once**('config.php');  
**if**($\_SERVER['REQUEST\_METHOD'] **==** "POST"){  
 *// Get data from the REST client* $task **=** **isset**($\_POST['task']) **?** mysqli\_real\_escape\_string($conn, $\_POST['task']) **:** "";  
 $date **=** **isset**($\_POST['date']) **?** mysqli\_real\_escape\_string($conn, $\_POST['date']) **:** "";  
 $priority **=** **isset**($\_POST['priority']) **?** mysqli\_real\_escape\_string($conn, $\_POST['priority']) **:** "";  
 *// Insert data into database* $sql **=** "INSERT INTO `my\_to\_do\_db`.`my\_to\_do\_tb` (`task`, `date`, `priority`) VALUES ('$task', '$date', '$priority');";  
 $post\_data\_query **=** mysqli\_query($conn, $sql);  
 **if**($post\_data\_query){  
 $json **=** **array**("status" **=>** 1, "Success" **=>** "To-Do has been added successfully!");  
 }  
 **else**{  
 $json **=** **array**("status" **=>** 0, "Error" **=>** "Error adding To-Do! Please try again!");  
 }  
}  
**else**{  
 $json **=** **array**("status" **=>** 0, "Info" **=>** "Request method not accepted!");  
}  
**@**mysqli\_close($conn);  
*// Set Content-type to JSON*header('Content-type: application/json');  
**echo** json\_encode($json);

**4. Create a PHP script**info.php**to fetch To-Do information from the list of To-Do's**

This script GETs the data from the MySQL database using task as the request query parameter. In other words, this script allows us to fetch a To-Do from the list using a To-Do task.

For example, let's assume we have a To-Do whose task name is **Write Code**. Now you can retrieve its information by using **Write Code**as a query string. More on it is covered later in this post.

**<?php**  
 **include\_once**('config.php');  
 $task **=** **isset**($\_GET['task']) **?** mysqli\_real\_escape\_string($conn, $\_GET['task']) **:** "";  
 $sql **=** "SELECT \* FROM `my\_to\_do\_db`.`my\_to\_do\_tb` WHERE task='{$task}';";  
 $get\_data\_query **=** mysqli\_query($conn, $sql) **or** **die**(mysqli\_error($conn));  
 **if**(mysqli\_num\_rows($get\_data\_query)**!=**0){  
 $result **=** **array**();  
   
 **while**($r **=** mysqli\_fetch\_array($get\_data\_query)){  
 extract($r);  
 $result[] **=** **array**("Task" **=>** $task, "Date" **=>** $date, 'Priority' **=>** $priority);  
 }  
 $json **=** **array**("status" **=>** 1, "info" **=>** $result);  
 }  
 **else**{  
 $json **=** **array**("status" **=>** 0, "error" **=>** "To-Do not found!");  
 }  
**@**mysqli\_close($conn);*// Set Content-type to JSON*header('Content-type: application/json');  
**echo** json\_encode($json);

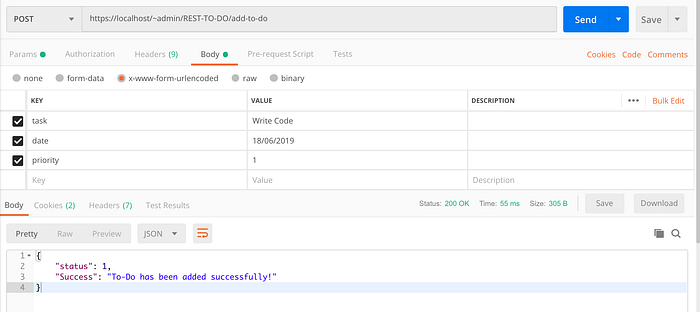
We’re now done with the coding. Time to see our API in action.

**Try creating a To-Do**

Now that you have written the code, it is time to test the API. For this, open your favorite REST client & send a POST call like below. You need to make sure you choose the **Body** as x-www-form-urlencoded & enter key-values accordingly. See the screenshot for a better understanding.

**REST endpoint**

**{**  
 https://localhost/~admin/REST-TO-DO/add-to-do  
**}**



If you have followed the steps correctly, you’ll see that your To-Do has been added to the database successfully! You should see an output similar to the below.

{  
 "status": 1,  
 "Success": "To-Do has been added successfully!"  
}

You can grab the entire code from my [GitHub Repository](https://github.com/shashank-ssriva/REST-To-Do). It also has the SQL script to create the database & its table. You can also click the link below to go to the repository.

**[shashank-ssriva/REST-To-Do](https://github.com/shashank-ssriva/REST-To-Do?source=post_page-----60ba1ad918d2--------------------------------" \t "_blank)**

[You can't perform that action at this time. You signed in with another tab or window. You signed out in another tab or…](https://github.com/shashank-ssriva/REST-To-Do?source=post_page-----60ba1ad918d2--------------------------------" \t "_blank)

[github.com](https://github.com/shashank-ssriva/REST-To-Do?source=post_page-----60ba1ad918d2--------------------------------" \t "_blank)

**Fetch To-Do information**

To retrieve a To-Do’s information, invoke a GET call against the below REST endpoint. Note how I am using ?task=Write Code as the request query parameter.

**REST endpoint**

**{**  
 https://localhost/~admin/REST-TO-DO/info?task**=**Write Code  
**}**

**Output**

{  
 "status": 1,  
 "info": [  
 {  
 "Task": "Write Code",  
 "Date": "18/06/2019",  
 "Priority": "1"  
 }  
 ]  
}

You have now reached the end of this tutorial. I hope it was informative & helpful. In my [next post, I will show you how you can add **Basic Authentication**](https://shashanksrivastava.medium.com/enable-basic-authentication-for-restful-web-service-2bd15f4384ba) to this API so that only authenticated users are able to add/fetch the To-Do’s.

I’ll also demonstrate how you can enable HTTPS for this API.

Thanks for reading this post!